



Application Case Studies with H5Part

Prabhat & Mark Howison
Lawrence Berkeley National Lab

HDF5 Workshop @ NERSC
January 21, 2009

Outline

- H5Part: Motivation
- Application: Laser Wakefield Analysis
- Application: GCRM Simulations
- Performance Issues

H5Part: Motivation

- HDF5
 - Rich, powerful, flexible API
 - Steep learning curve
- H5Part
 - Customized for particle accelerator community
 - Simplified, veneer API:
C, C++, F77, Python bindings
 - Ported to all modern HPC platforms
 - Open source (BSD-like license)

<http://www-vis.lbl.gov/Research/AcceleratorSAPP/>

<https://codeforge.lbl.gov/projects/h5part/>

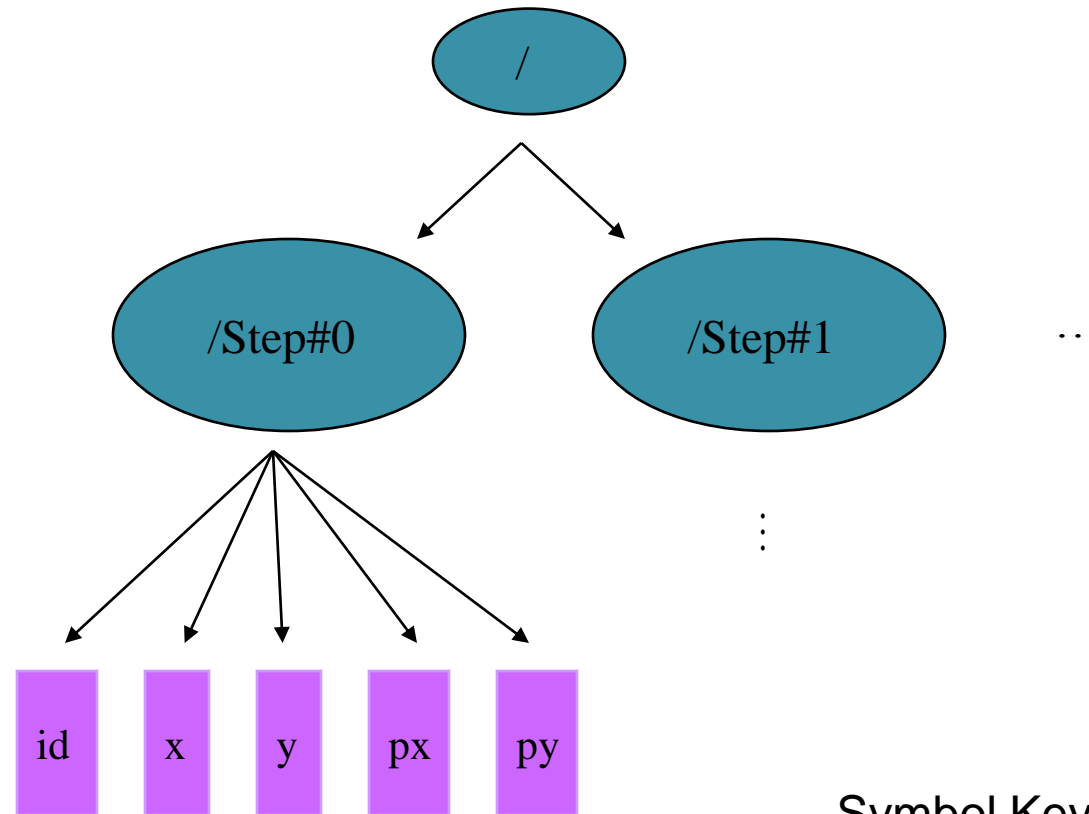
H5Part: Sample code



```
#include "mpi.h"
#include "H5Part.h"
H5PartFile *file;
file = H5PartOpenFileParallel(
    "test.h5",
    H5PART_WRITE,
    MPI_WORLD_COMM);

for (int t=0; t<timesteps; t++) {
    H5PartSetStep(file,t);
    H5PartSetNumParticles(file,sz);
    // do the simulation
    H5PartWriteDataFloat64(file,"x",x);
    H5PartWriteDataFloat64(file,"px",px);
}

H5PartCloseFile(file);
```

H5Part file



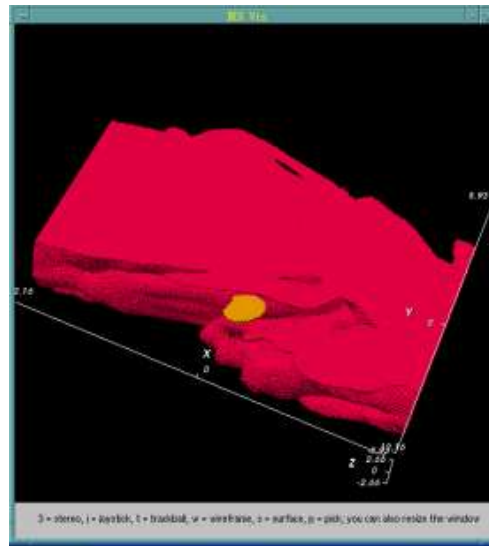
- Symbol Key
-  *HDF5 Group* contains subgroups, datasets and attributes
 -  *HDF5 Dataset* contains data arrays

FastBit

- Developed by John Wu (LBNL SDM Center)
- State-of-the-art index/query system
- Processes range queries in time \sim #hits
- Don't have to load the entire dataset!

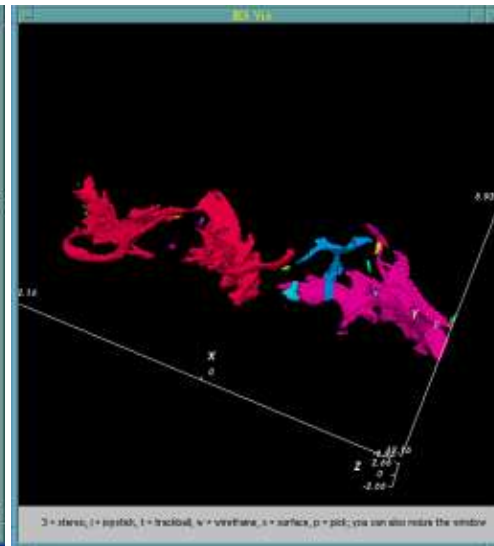
Query

CH4 > 0.3



Query

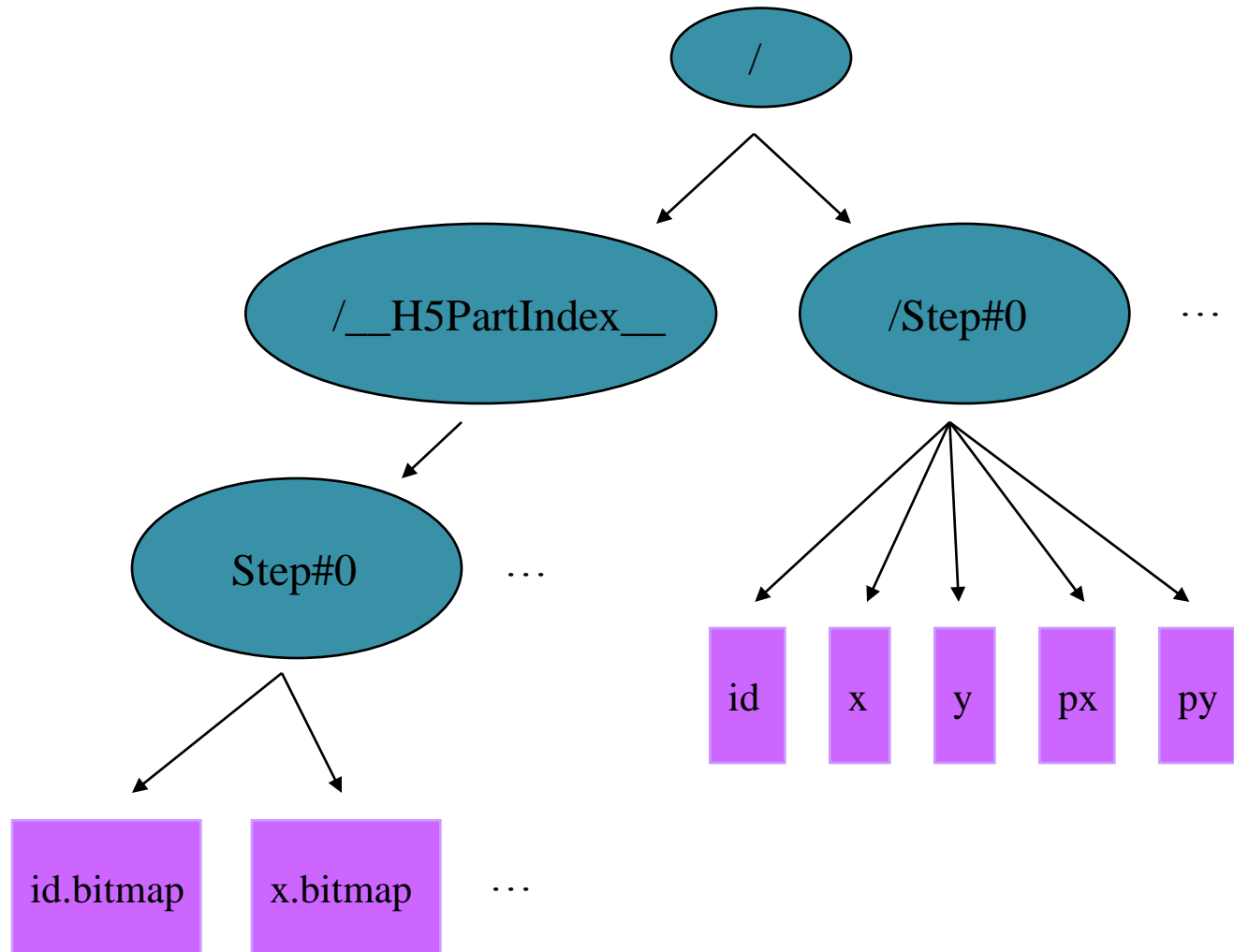
CH4 > 0.3
&&
temp < 3



<https://codeforge.lbl.gov/projects/fastbit>

<http://sdm.lbl.gov/fastbit/>

H5Part file with FastBit keys





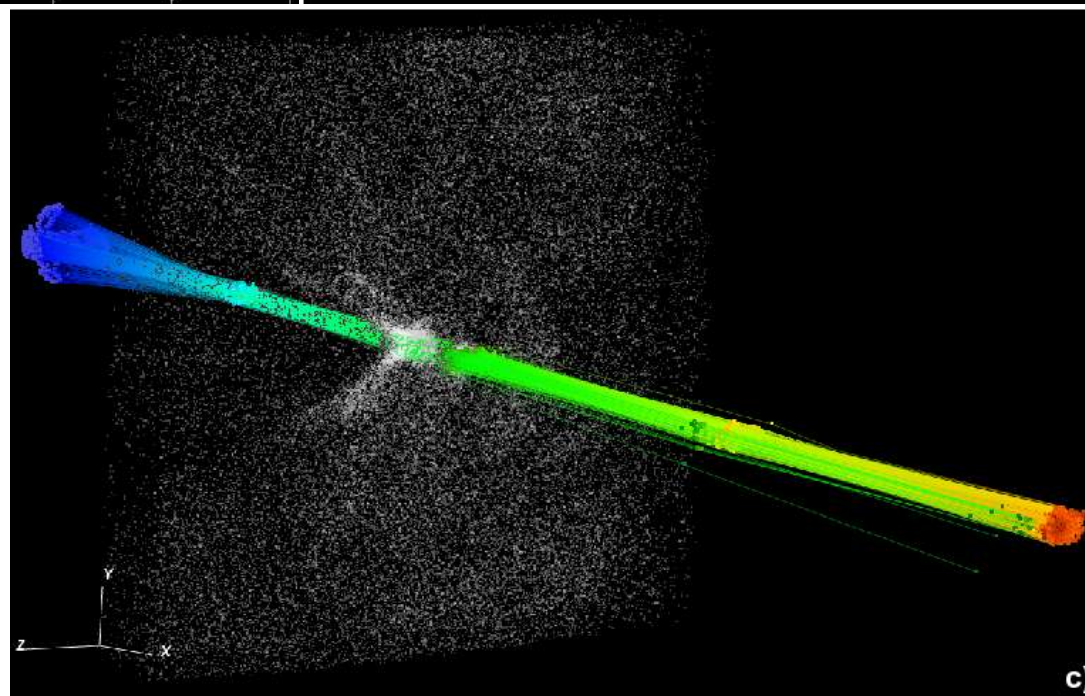
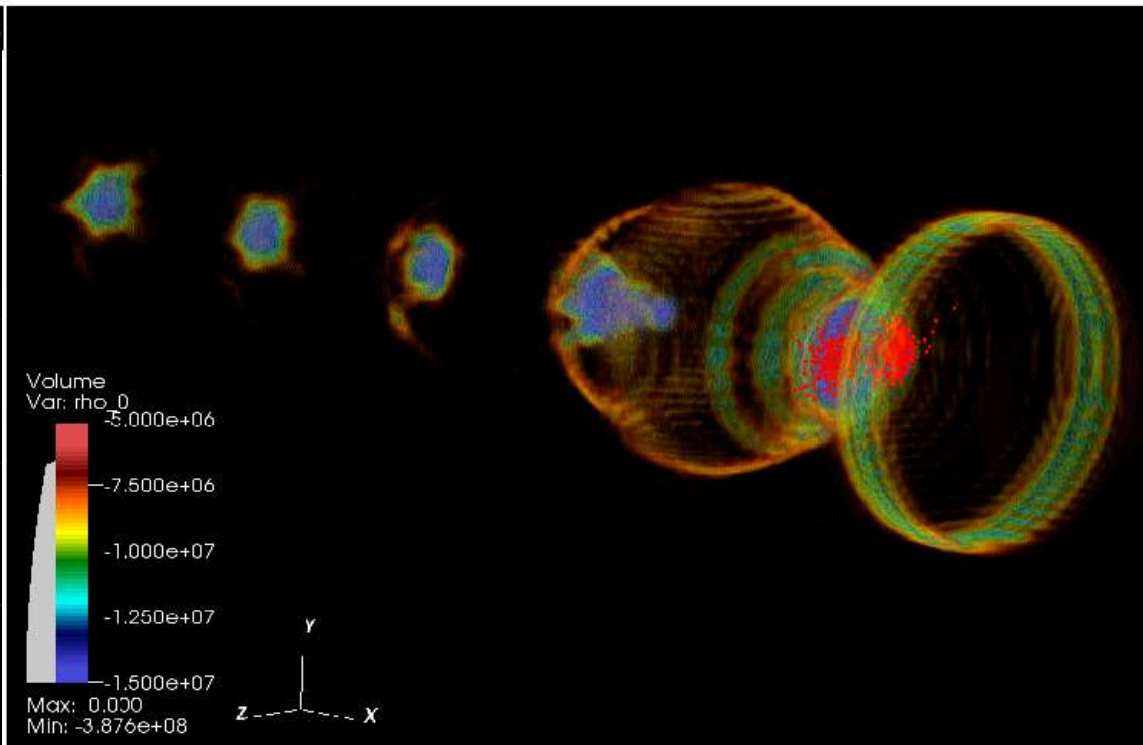
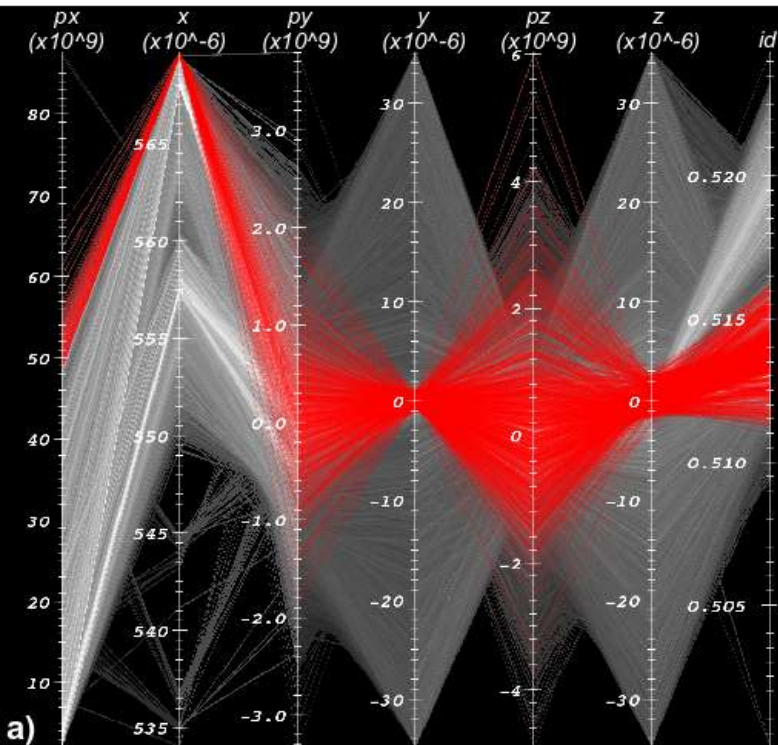
Application to Laser Wakefield Analysis

Laser Wakefield Analysis

- VisIt plugin to load FastBit-enhanced H5Part data
- Rapid generation of parallel co-ordinate plots
- Interactive specification of queries, e.g.
`px>1e10 && y>0`
- Tracking of particles across time
- Tracks 500 particles in 1.5TB of data in 0.15 seconds
 - IDL scripts take ~2.5 hours on smaller 5GB dataset

SC08 paper: “*High Performance Multivariate Visual Data Exploration for Extremely Large Data*”

Oliver Rübel, Prabhat, Kesheng Wu, Hank Childs, Jeremy Meredith, Cameron G.R. Geddes, Estelle Cormier-Michel, Sean Ahern, Gunther H. Weber, Peter Messmer, Hans Hagen, Bernd Hamann and E. Wes Bethel



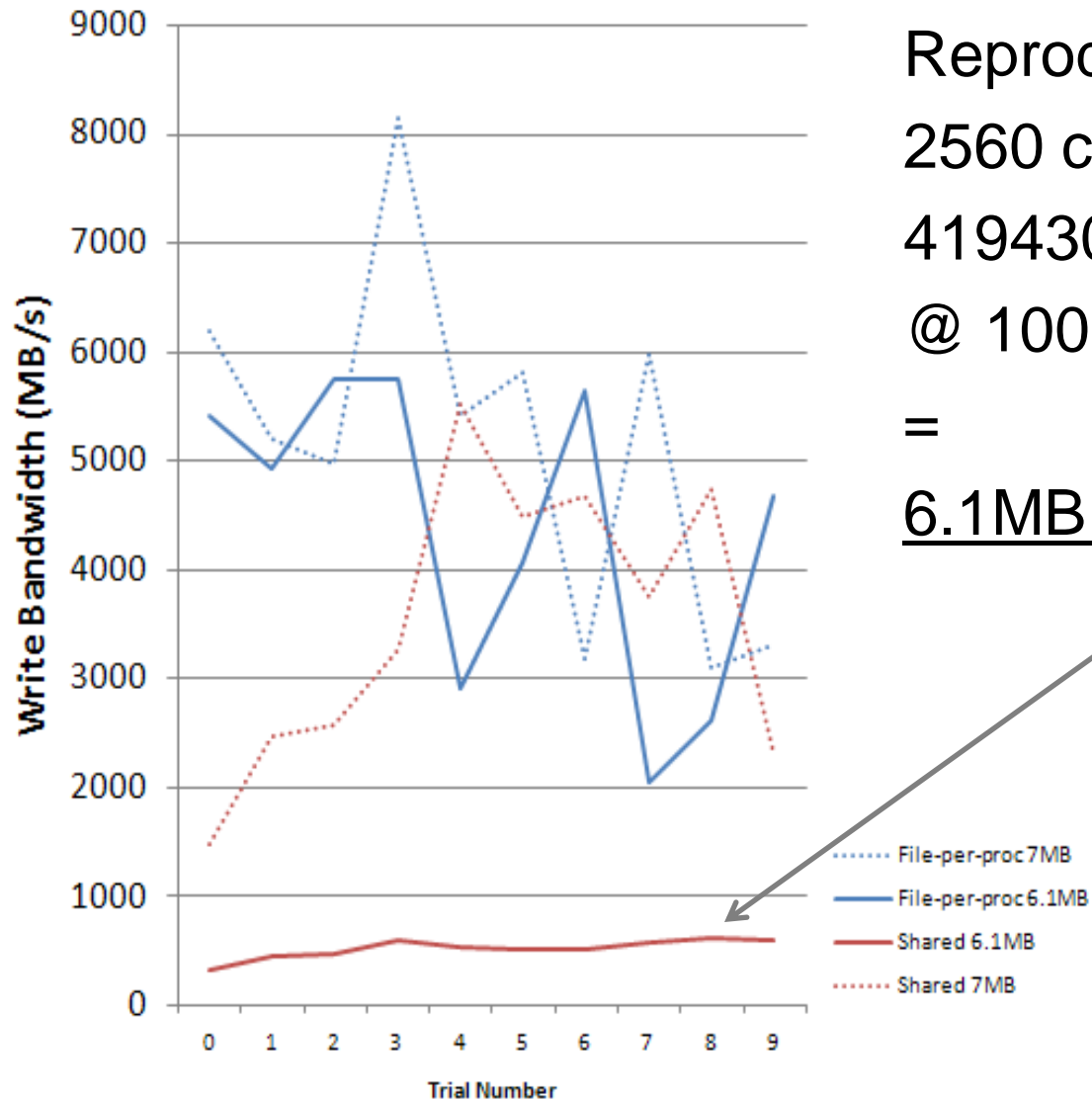


Application to Global Cloud Resolving Model simulations

GCRM Project

- Dave Randall, SciDAC/INCITE
 - 3.9 km resolution model
 - 24 hour run on 30K nodes
 - Generates 10TB of data
 - Sustained 2GB/s write performance required for IO to take <5% runtime

GCRM IO pattern



Reproduced in IOR

2560 core test run

41943042 cells

@ 100 levels

=

6.1MB written per core

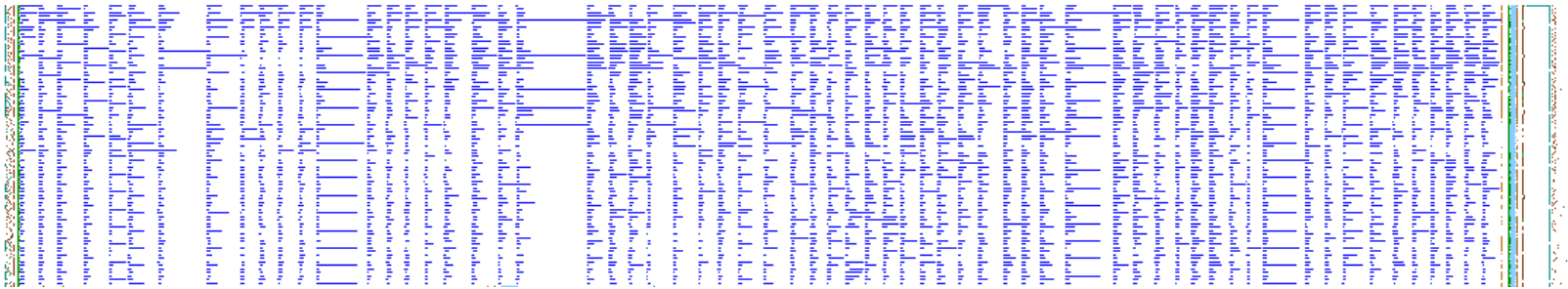
Performance issues (system)

- < 1GB/s write bandwidth when IO patterns do not align to lustre stripes
- Shared file performance is worse than file-per-proc, except in special cases
- MPI-IO collective mode (2-phase) is effectively broken in vendor library

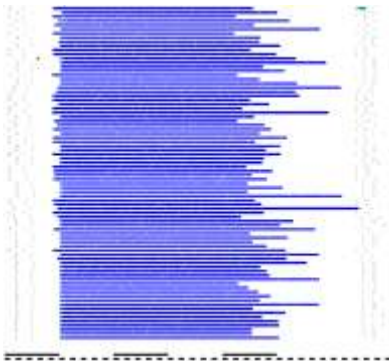
Performance issues (MPI-IO)

Synchronous vs. Asynchronous Write Calls for Same IO Pattern

Cray's MPI-IO Implementation (1294 MB/s) ~ MPI-IO VFD collective mode



IOR POSIX Shared File (6535 MB/s) ~ MPI-POSIX VFD



Test Parameters

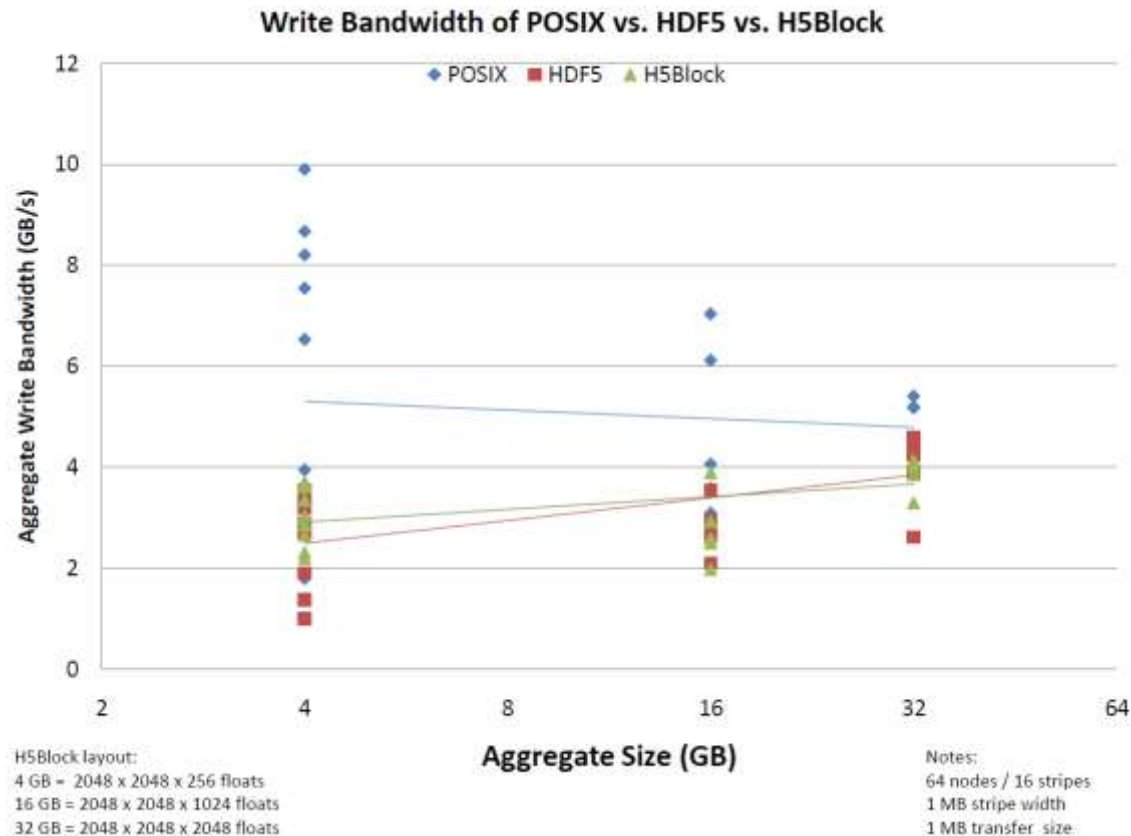
Nodes/stripes: 80
Aggregate data: 40GB
Stripe width: 8MB
Write size: 8MB
Writes per node: 64

Key

Open
Read
Write
Seek
Close

Performance issues (HDF5)

- H5Block and HDF5 (MPI-POSIX VFD) performance is close to POSIX Shared File

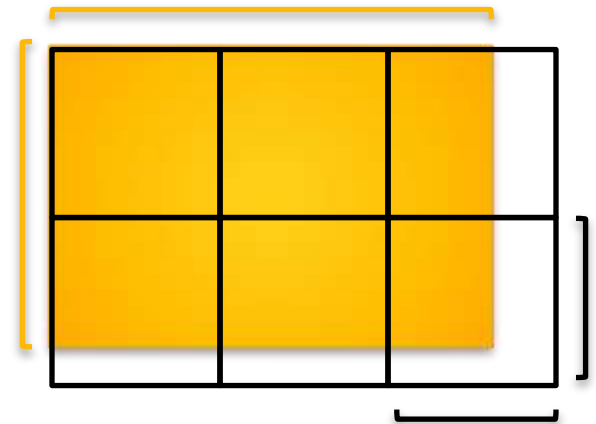


Performance issues (HDF5)

- Can only use chunk + pad in one dimension
 - How do you chunk + pad a 3D array when none of the dimensions are multiples of the 1MB stripe width? E.g. 100^3 ?
 - Splitting arbitrary dimensional array into contiguous 1MB pieces is difficult
 - 2-phase may be only viable solution:

- Aggregate 3D array on writer nodes
- Writer node treats data as flat 1D array, splits into 1MB chunks

array size



chunk size

Action items for HDF5 team

- Make HDF5 lustre aware
 - Add lustre hooks to HDF5 tunable parameters
 - Pad/align chunks to stripe boundaries
 - Handle arbitrary array sizes?
- Enable 2-phase IO functionality
 - Fewer writer nodes reduces burden on OSTs
 - Data shipping leverages SeaStar bandwidth
 - User space solutions are complicated: want solution at middle-ware level (e.g. MPI-IO or HDF5)
 - Possible implementations:
 - Add to HDF5 MPI-POSIX VFD
 - Wait for Cray improvements to MPI-IO library and use MPI-IO VFD

Acknowledgements

- H5Part + FastBit
 - Andreas Adelman, Achim Gsell (PSI)
 - John Shalf, Wes Bethel, Kurt Stockinger, John Wu (LBNL),
 - Luke Gosink (UCDavis)
- LWFA
 - Cameron Geddes, Estelle Michel (LBNL)
 - Peter Messmer (Tech-X)
- GCRM
 - DOE Office of Science (BER)
 - Micheal Wehner, John Shalf (LBNL/NERSC)
 - Wes Bethel, VACET (LBNL/NERSC)
 - Dave Randall, Ross Heikes (Colorado State Univ)
 - Karen Schuchardt, Bruce Palmer, Annette Koontz (PNNL)
- IPM
 - Noel Keen (LBNL)